

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing Of Claims:

Please amend the claims as follows:

1. (Currently Amended) A system for offloading an input/output (I/O) task from a first computer to a second computer, comprising:
 - a client running on the first computer;
 - a server running on the second computer; and
 - at least one remote direct memory access (RDMA) channel linking the first computer and the second computer, wherein the first computer and the second computer communicate in accordance with a protocol comprising:
 - a network discovery phase, wherein the network discovery phase is configured to:
 - create, by the client, an RDMA connection to the server over a shared RDMA-capable provider; and
 - authenticate, by the client and the server, the RDMA connection,
 - wherein the network discovery phase configured to authenticate the RDMA connection is configured to:
 - send, by the server, a credit request message, wherein the credit request message comprises one of the following: the number

of credits the client have to give up and the number of credits that the server has newly allocated for use by the client;

receive, by the client, the credit request message;

in response to the client receiving the credit request message, send, from the client to the server, one of the following:

one message if an information field in the credit request message is positive, wherein the positive credit request message comprises credits that the server has newly allocated for use by the client, and

at least one message if the information field in the credit request message is negative, wherein the negative credit request message comprises one message for each credit that the client has to give up; and

in response to receiving the message from the client, send by the server, a response for each message received from the client, and

an I/O processing phase, wherein read operations are implemented using RDMA and write operations are implemented using send operations, wherein the write operations are not implemented using RDMA.

2. (Canceled)

3. (Original) The system of claim 1 wherein the protocol is used in association with a second network protocol.

4. (Previously Presented) The system of claim 3 wherein the second protocol is a server message block (SMB).

5. (Previously Presented) The system of claim 3 wherein the second protocol is a common internet file system (CIFS).

6. (Currently Amended) A computer-readable medium storing computer-executable instructions and computer-readable data comprising a computer program product for use in a system for offloading an input/output (I/O) task from a first computer to a second computer, the system comprising:

at least one remote direct memory access (RDMA) channel linking the first computer and the second computer, wherein the first computer and the second computer communicate in accordance with a protocol comprising:

a network discovery phase, wherein the network discovery phase is configured to:

create, by the client, an RDMA connection to the server over a shared RDMA-capable provider; and

authenticate, by the client and the server, the RDMA connection, wherein the network discovery configured to authenticate the RDMA connection is configured to:

send, by the server, a credit request message, wherein the credit request message comprises one of the following: the number of credits the client have to give up and the number of credits that the server has newly allocated for use by the client;

receive, by the client, the credit request message;

in response to the client receiving the credit request message, send, from the client to the server, one of the following:

one message if an information field in the credit request is positive, wherein the positive credit request message comprises credits that the server has newly allocated for use by the client, and

at least one message if the information field in the credit request is negative, wherein the negative credit request message comprises one message for each credit that the client has to give up; and

in response to receiving the message from the client, send, by the server, a response for each message received from the client; and

an I/O processing phase, wherein read operations are implemented using RDMA and write operations are implemented using send operations, wherein the write operations are not implemented using RDMA.

7. (Currently Amended) A method for offloading an input/output (I/O) task from a first computer to a second computer, comprising:

discovering, by a client on the first computer and a server on the second computer, at least one shared remote direct memory access (RDMA) capable provider, wherein discovering comprises:

creating, by the client, an RDMA connection to the server over the at least one shared RDMA-capable provider; and

authenticating, by the client and the server, the RDMA connection,

wherein authenticating the RDMA connection comprises:

sending, by the server, a credit request message, wherein the credit request message comprises one of the following: the number of credits the client have to give up and the number of credits that the server has newly allocated for use by the client,

receiving, by the client, the credit request message,

in response to the client receiving the credit request message,

sending, from the client to the server, one of the following:

one message if an information field in the credit request message is positive, wherein the positive credit request message comprises credits that the server has newly allocated for use by the client, and

at least one message if the information field in the credit request message is negative, wherein the negative credit request

message comprises one message for each credit that the client has
to give up; and
in response to receiving the message from the client, sending, by
the server, a response for each message received from the client; and
posting, by the client, an I/O processing request for completion by the server on
the second computer, wherein read operations are implemented using RDMA and write
operations are implemented using send operations, wherein the write operations are not
implemented using RDMA.

8. (Previously Presented) The method of claim 7 wherein the discovering at
least one shared RDMA-capable provider further comprises:
obtaining, by the client, a server request resume key from the server;
opening, by the client, a pipe to the server;
sending, by the client over the pipe, a negotiate request; and
sending, by the server over the pipe, a negotiate response including a minimal list
of common providers.

9. (Cancelled)

10. (Currently Amended) The method of claim [[9]] 7, further comprising:
registering, by the client, one or more files for use with the server over the RDMA
connection.

11. (Previously Presented) The method of claim 10 wherein the registering at least one file comprises:

sending, by the client to the server, a register file message; and

sending, by the server to the client, a register file completion message.

12. (Currently Amended) The method of claim ~~[[9]]~~ 7, wherein the authenticating the RDMA connection further comprises:

sending, by the client, an authenticate request message to the server, the authenticate request message including a key;

if the key matches a previous key sent by the server to the client, sending, by the server, an authenticate response message to the client.

13. (Original) The method of claim 12 wherein the previous key is a key contained in a negotiate response message sent by the server to the client.

14. (Original) The method of claim 12, further comprising:

sending, by the server to the client, a status response message to complete the authenticating.

15. (Original) The method of claim 7 wherein the posting the I/O processing request comprises sending, by the client, one of (a) a close request, (b) a cancel request, (c) a read request, (d) a write request, (e) a vectored read request, and (f) a vectored write request.

16. (Original) The method of claim 15, further comprising:
completing, by the server, the read request and the vectored read request by
sending data using RDMA write operations; and
completing, by the server, the write request and the vectored write request by
sending data using normal send operations.
17. (Original) The method of claim 15 wherein the vectored write request
includes a collapse flag in a header of the request.
18. (Original) The method of claim 7 wherein posting the I/O processing
request further includes indicating whether the completion by the server should be in
polling mode.
19. (Original) The method of claim 18 wherein the indicating whether the
completion should be in polling mode comprises indicating that the completion should
not be in polling mode by setting an interrupt flag in a header of the I/O processing
request.
20. (Original) The method of claim 18, further comprising:
if the client indicates that the completion should not be in polling mode,
completing, by the server, the I/O processing request by sending a status block to the first
computer by way of RDMA transfer.

21. (Original) The method of claim 18, further comprising:

if the client indicates that the completion should be in polling mode, and the client has sent an interrupt request message to the server, sending, by the server to the client, an interrupt response message by way of an ordinary send.

22. (Original) The method of claim 7 wherein posting the I/O processing request further includes specifying a number of credits in a header of the request.

23. (Currently Amended) A computer-readable media storing computer-executable instructions for implementing a method for offloading an input/output (I/O) task from a first computer to a second computer, the method comprising:

discovering, by a client on the first computer and a server on the second computer, at least one shared remote direct memory access (RDMA) capable provider,

wherein the first computer requests a server request resume key and the second computer passes the server request resume key as an authentication mechanism, wherein requesting and passing the request resume key comprises:

creating, by the client, an RDMA connection to the server over the at least one shared RDMA-capable provider; and

authenticating, by the client and the server, the RDMA connection, wherein authenticating the RDMA connection comprises:

sending, by the server, a credit request message, wherein the credit request message comprises one of the following: the

number of credits the client have to give up and the number of credits that the server has newly allocated for use by the client: receiving, by the client, the credit request message; in response to the client receiving the credit request message, sending, from the client to the server, one of the following:

one message if an information field in the credit request message is positive, wherein the positive credit request message comprises credits that the server has newly allocated for use by the client, and

at least one message if the information fielding the credit request message is negative, wherein the negative credit request message comprises one message for each credit that the client has to give up; and

in response to receiving the message from the client, sending, by the server, a response for each message received from the client;

wherein after authentication, the first computer opens a pipe to the second computer and queries the second computer for a list of shared RDMA-capable providers; and

posting, by the client, an I/O processing request for completion by the server on the second computer, wherein read operations are implemented using RDMA and write

operations are implemented using send operations, wherein the write operations are not implemented using RDMA.

24. (Withdrawn)